

SELF-REPAIRING REDUNDANCY FOR MEMORY BLOCKS IN PROGRAMMABLE
LOGIC DEVICES

FIELD OF THE INVENTION

[0001] The invention relates to Programmable Logic Devices (PLDs). More particularly, the invention relates to a memory block in a PLD that detects and automatically repairs defects during the configuration process for the PLD.

BACKGROUND OF THE INVENTION

[0002] Programmable logic devices (PLDs) are a well-known type of integrated circuit that can be programmed to perform specified logic functions. One type of PLD, the field programmable gate array (FPGA), typically includes an array of configurable logic blocks (CLBs) and programmable input/output blocks (IOBs). The CLBs and IOBs are interconnected by a programmable interconnect structure.

[0003] The logic blocks and interconnect structure are typically programmed by loading a stream of configuration data (bitstream) into internal configuration memory cells that define how the logic blocks and interconnect are configured. The configuration data can be read from memory (e.g., an external PROM) or written into the FPGA by an external device. The collective states of the individual memory cells then determine the function of the FPGA.

[0004] Some FPGAs also include additional logic blocks with special purposes in the configurable array. For example, the Xilinx Virtex-II Pro™ FPGA includes blocks of Random Access Memory (RAM), blocks implementing multiplier functions, and embedded processor blocks. (The Xilinx Virtex-II Pro FPGA is described in detail in pages 19-71 of the "Virtex-II Pro Platform FPGA Handbook", published October 14, 2002 and available from Xilinx, Inc., 2100 Logic Drive, San Jose, California 95124, which pages are incorporated herein by reference.)

[0005] FPGAs are increasing in size with each product generation. At the same time, the minimum feature size (e.g., gate length, metal width, and so forth) is decreasing. Each of these trends increases the susceptibility of the FPGA to manufacturing defects. Therefore, it is desirable to provide structures and methods that reduce the susceptibility of FPGA circuits to manufacturing defects. It is particularly desirable to reduce the susceptibility of repetitive circuits to manufacturing defects. For example, if a logic block that is repeated many times is rendered less susceptible, the gain in product yield is larger than the gain resulting from a similar improvement to a circuit that is included only once. Further, it is particularly desirable to reduce the susceptibility of circuits that can be unusually sensitive to manufacturing defects, such as memory cells in a RAM array that typically comprise minimum-sized transistors designed to consume the smallest possible area.

[0006] A known method of reducing susceptibility to manufacturing defects in a PLD RAM array is to include a redundant column of memory cells in the RAM array. For example, such a method is described by Cliff et al. in U.S. Patent No. 5,498,975 entitled "Implementation of Redundancy on a Programmable Logic Device", which is hereby incorporated herein by reference. When this type of PLD RAM array is used, each column of memory cells is tested during the PLD testing process. If a defect is detected in any column in the array, fuses are blown to effectively bypass the column of RAM cells. The device then functions with a column offset in the array, but in a manner transparent to the PLD user. In some cases, non-volatile memory cells (e.g., EEPROM or flash cells) are used to control the column selection instead of fuses.

[0007] However, this type of device has its disadvantages. For example, the fuses or non-volatile memory cells must be manufactured as part of the PLD, which adds to the manufacturing complexity and cost. An additional

disadvantage of fuse-based methods is that the testing and repair are typically performed "at the factory" (prior to shipping to the user), and any column offset is made permanent at that time. If a defect develops (i.e., one or more of the RAM columns ceases to operate correctly) after the column offset procedure is performed and the device is already in place within the target system, it is then too late to repair the defect. Even if non-volatile memory cells are used, reprogramming of the non-volatile memory cells typically involves the use of a separate piece of programming hardware that might be unavailable within the user's system.

[0008] Therefore, it is desirable to provide methods and structures for self-test and self-repair of PLD RAM arrays.

SUMMARY OF THE INVENTION

[0009] The invention provides a programmable logic device (PLD) including a self-repairing RAM circuit, and a method of configuring the PLD. A PLD RAM circuit including one or more redundant columns is tested during the PLD configuration sequence using a built in self test (BIST) procedure. If a defective column is detected (e.g., the read data does not match the write data previously written to the column), an error flag associated with the column is set, using a volatile memory circuit to store the error flag. After the BIST procedure is complete, the PLD configuration process continues by loading the configuration data into the PLD, including the RAM circuit. The presence of the error flag causes the configuration data to bypass the defective column and to be passed directly into a replacement column (e.g., the adjacent column). The configuration of the RAM circuit continues until the remainder of the circuit is configured, including the redundant column. Thus, the detection and replacement of the defective column are transparent to the PLD user.

[0010] In some embodiments, multiple redundant columns are included in the RAM circuit. The multiple redundant columns

can be adjacent to each other, spaced equally within the RAM circuit, grouped such that the redundant columns are adjacent and the groups are equally spaced, or in some other configuration. In some embodiments, the PLD includes two or more similar RAM circuits (e.g., configurable RAM blocks in an FPGA), and a BIST control circuit provides control signals simultaneously to the two or more similar RAM circuits. Thus, the similar RAM circuits are tested in parallel by a single BIST control circuit.

[0011] According to another aspect of the invention, the BIST procedure is initiated independently of the configuration sequence for the PLD. For example, in a PLD system operating continuously "in the field", the BIST procedure can be initiated at any time on a previously-configured PLD. If a defective column is detected in the PLD (e.g., the read data does not match the write data previously written to the column), an error flag associated with the column is set, using a volatile memory circuit to store the error flag. When user operation of the PLD resumes, write data directed to the defective column is passed into a replacement column, and read data from the replacement column is provided in place of read data from the defective column.

BRIEF DESCRIPTION OF THE DRAWINGS

[0012] The present invention is illustrated by way of example, and not by way of limitation, in the following figures.

[0013] Fig. 1 is a block diagram of a first PLD including a RAM circuit that is automatically tested and repaired during the PLD configuration process, according to an embodiment of the present invention.

[0014] Fig. 2 is another block diagram of the PLD of Fig. 1 that illustrates the usage of a defective RAM circuit, according to an embodiment of the invention.

[0015] Fig. 3 is a more detailed view of the routing circuit included in the PLD of Fig. 1.

[0016] Fig. 4 illustrates an exemplary volatile memory circuit that can be used, for example, in the routing circuit of Fig. 3.

[0017] Fig. 5 is an exemplary state diagram for the BIST control circuit included in the embodiment of Fig. 1.

[0018] Fig. 6 is a block diagram of a second PLD including two RAM circuits that are automatically tested and repaired during the PLD configuration process according to an embodiment of the present invention.

[0019] Fig. 7 illustrates the steps of an exemplary method of configuring a PLD, according to another aspect of the present invention.

[0020] Fig. 8 illustrates the steps of a method of detecting and repairing defective columns in a RAM array, according to another embodiment of the invention.

DETAILED DESCRIPTION OF THE DRAWINGS

[0021] In the following description, numerous specific details are set forth to provide a more thorough understanding of the present invention. However, it will be apparent to one skilled in the art that the present invention can be practiced without these specific details.

[0022] Fig. 1 is a block diagram of a first PLD including a RAM circuit that is automatically tested and repaired during the PLD configuration process, according to an embodiment of the present invention. The PLD of Fig. 1 includes a RAM circuit 101, a BIST control circuit 102, a routing circuit 103, a configuration control circuit 104, a configuration data port 105, and, optionally, additional PLD logic 106. RAM circuit 101 includes an array 108 comprising n columns of RAM cells provided on bit lines B1-B n , where n is a positive integer. However, RAM circuit 101 also includes an additional column 109 of RAM cells referred to herein as the redundant column. The redundant column is provided as a replacement for any column in array 108 that might prove defective.

[0023] Configuration control circuit 104 is coupled to receive configuration data from configuration data port 105, and to supply the configuration data to RAM circuit 101 (as part of signals B) and to other PLD logic 106 (as part of signals A). Another function of configuration control circuit 104 is to provide control signals via signals A, B, and D to the additional PLD logic 106, RAM circuit 101, and BIST control circuit 102, respectively. (Note that in the present specification, the same reference characters are used to refer to terminals, signal lines, and their corresponding signals.)

[0024] With the exception of logic associated with signals D, configuration control circuit 104 can be implemented, for example, using well known circuits such as those used in the aforementioned Virtex-II Pro FPGAs from Xilinx, Inc. Signals D can include, for example, a configuration clock signal and one or more enable signals provided by configuration control circuit 104 to BIST control circuit 102. The enable signal initiates the BIST procedure used by BIST control circuit 102 to test array 108, and the configuration clock signal is used to clock the BIST procedure. Signals D can also include, for example, status signals provided by BIST control circuit 102 to configuration control circuit 104. Such a status signal can be used, for example, to indicate the termination of the BIST procedure.

[0025] BIST control circuit 102 is a state machine that generates the test data patterns, read/write addresses, and control signals needed to perform one or a series of tests on array 108. (An exemplary state diagram for the BIST state machine is shown in Fig. 5 and is described below in conjunction with that figure.) The read and write addresses are generated using binary counters, which are well known. Data loaded into array 108 preferably follows the well known checkerboard and reverse checkerboard patterns. For example, the pattern 0101_1010_0101_1010... can be used, followed by the pattern 1010_0101_1010_0101.... These patterns are

easily generated using known logic circuits. Each of these patterns is first written to and then read back from array 108 to detect stuck-at and bridging faults. BIST control circuits utilizing checkerboard and reverse checkerboard patterns are well known in the relevant arts. In other embodiments, patterns other than the checkerboard and reverse checkerboard patterns are used to provide the desired test coverage.

[0026] Routing circuit 103 illustrates how the redundant column 109 of RAM cells can be used in the event of a defective column in array 108. Multiplexers M1-Mn select either the bit (Bi) from the associated column, or the bit (Bi+1) from the adjacent column. (Note that multiplexers M1-Mn function as multiplexers during a read from the RAM circuit, and as demultiplexers during a write to the RAM circuit.) If a defective column is found, the multiplexer Mi associated with the defective column is configured to select the bit (Bi+1) from the adjacent column. All multiplexers to the right of the defective column are also configured to select the bits from the adjacent column.

[0027] Fig. 2 illustrates the portions of RAM circuit 101 that will be used when array 108 includes an exemplary defective RAM cell. In the pictured embodiment, RAM cell 210 is defective. (The defective circuitry is denoted by a cross-hatched pattern.) Therefore, the column of RAM cells that includes RAM cell 210 is bypassed both during the configuration of the RAM array with the configuration data stream, and during later user operation of the configured device. Instead, routing circuit 103 selects the column to the right of the column including RAM cell 210, and this shifting applies for all of the remaining columns. Thus, the redundant column 109 is used to interface to the final bit Bitn. The RAM cells that are actually configured by the subsequent configuration process, and then are available for use by a user design, are denoted in Fig. 2 by a fill pattern of diagonal lines.

[0028] Fig. 3 shows additional details of routing circuit 103, and illustrates how the column bypass is achieved. Multiplexers M1-Mn are each controlled by an associated AND gate A1-An, which is driven by a carry-in signal from the left and a signal from an associated volatile memory circuit VM1-VMn. When one of volatile memory circuits VM1-VMn stores a low value, the associated AND gate A1-An drives a low value. This low value controls the associated multiplexer Mi to select the bit signal Bi+1 from the adjacent column rather than signal Bi from the associated column of RAM cells. Further, the low value is also passed to the adjacent AND gate to the right, which again forces the associated multiplexer Mi+1 to select the bit signal Bi+2 from the adjacent column. This process continues along the chain from AND gate to AND gate, ensuring that all subsequent column bits are shifted by one.

[0029] As logically illustrated in Fig. 3, the values stored in volatile memory circuits VM1-VMn are initially set (high). If the BIST procedure detects a defective column, the value stored in the associated volatile memory circuit is reset (low), and that column is bypassed using the process described above. In other embodiments, the values stored in the volatile memory circuits are initially reset, and are set if a defective column is detected. In either case, each AND gate Ai is initially configured to select the associated bit signal Bi, then is reconfigured to select the adjacent bit signal Bi+1 if an error is detected. In either case, the change in the value stored in the volatile memory circuit is referred to herein as "setting an error flag".

[0030] Fig. 4 illustrates one embodiment of volatile memory circuits VM1, VM2, VM3... that can be used, for example, in the routing circuit of Fig. 3. Cross-coupled inverters 402, 403 and transistors 406, 407 together form a volatile RAM cell. The value stored in the volatile RAM cell is denoted in Fig. 4 as node Q. Node Q is initially low, and signal Data Out (the data output terminal of the volatile memory circuit) is high, selecting the associated column in

the RAM circuit. Provided that signal BIST_done is high, a high value on node Q causes NAND gate 401 to provide a low value to the Data Out terminal. As previously described in connection with Fig. 3, a low value on the Data Out terminal causes the associated column of the RAM circuit to be bypassed. Note that if signal BIST_done is low, signal Data Out is always high, and the associated column is not bypassed. The BIST_done signal is provided by the BIST control circuit (see Fig. 1), and is used to prevent the bypassing of any columns until after the BIST procedure is complete.

[0031] Exclusive NOR (XNOR) gate 405 performs the comparison between the data written to the array (Wdata) with the data read back from the same location (Rdata). If the data differ, signal EQ_b is high, enabling transistor 409. When transistor 408 is enabled by write enable signal WE, node QB (which is initially high) is pulled to a low value while node Q goes high. Thus, as described above, the associated column is bypassed once signal BIST_done goes high.

[0032] An exemplary BIST procedure includes the following series of steps:

- 1) generating write and read addresses using binary counters;
- 2) enabling the write and read controls;
- 3) writing the test data to the array;
- 4) reading back from the array the data from the same address;
- 5) comparing the read data with the previous write data; and
- 6) loading the result to a volatile memory cell associated with the write and read address.

[0033] Note that step 6 is the step in which the error flag is set for a defective array address. Note further that these steps are not necessarily performed sequentially. For

example, in one embodiment the write address is advanced by 16 addresses compared to the read address, then after that point the writes, reads, and compares occur concurrently. After all the writes are complete, 16 more reads and compares occur. After all the comparisons are complete, the results are loaded into an associated series of volatile memory cells.

[0034] To perform these steps, an exemplary BIST control circuit includes three major components: read and write pointer circuits; a data pattern generator; and a control block (e.g., a state machine). Read and write pointer circuits are well known, as are data pattern generators such as those providing checkerboard and reverse checkerboard patterns. Any appropriate implementation of these circuits can be used to implement the structures and methods of the present invention. Therefore, these circuits are not described in detail herein.

[0035] The control circuit is best understood in conjunction with the state diagram shown in Fig. 5. The state machine of Fig. 5 can be used, for example, with the embodiment illustrated in Fig. 1. Briefly, the states shown in Fig. 5 are as follows:

- Start: Start State
- State 1: Write first M data values using checkerboard pattern, where M is an integer
- State 2: Begin reads and compares; continue writes. Continue until all reads and compares are complete. Set error flags in volatile memory cells when errors are found.
- State 3: Reset read and write pointers. Write first M data values using reverse checkerboard pattern
- State 4: Begin reads and compares; continue writes. Continue until all reads and compares are complete. Set error flags in volatile memory cells when errors are found.
- Done: Enable column bypassing

[0036] The Start State 501 is entered when a redundancy self repair (RSR) signal Start_RSR goes high. The Start_RSR signal can be driven high, for example, by the configuration control circuit when the circuit is powered up. Additionally or alternatively, the RSR signal can be driven high by an external signal, e.g., an enable signal provided by a user.

[0037] From the Start State 501, the BIST control circuit enters State 1 (502) wherein the write counter is enabled for a limited period (e.g., 16 clock cycles) and test data in a checkerboard pattern is written to a sequence of addresses in the array. This step ensures the completion of the write step prior to attempting read back of the data stored at any given address. When this predetermined point is reached (e.g., the write address equals 15, or Wptr=15), the BIST control circuit enters State 2 (503).

[0038] In State 2 (503), the read counter is enabled. The write counter is also still enabled, therefore, writes continue to sequential addresses. Simultaneously, test data is read from sequential addresses in the array and compared to the write data for the corresponding address. If a discrepancy is found, a flag is set by changing a value stored in a volatile memory circuit associated with the write and read address. Eventually, the read pointer reaches a predetermined maximum value, e.g., test data has been read from the highest supported address. In one embodiment, there are 512 addresses in the array, so State 2 continues until the read pointer reaches a value of 511. This condition is indicated in Fig. 5 as "Rptr=511". When this point is reached, the BIST control circuit enters State 3 (504).

[0039] In States 3 and 4 (504 and 505, respectively, in Fig. 5), the write and read pointers are reset, and the process described for States 1 and 2 is repeated using a reverse checkerboard pattern. The combination of checkerboard and reverse checkerboard patterns is commonly used to detect stuck-at-high, stuck-at-low, and bridging

defects. In other embodiments, other test patterns are used. In some embodiments, only one test pattern is used. In some embodiments, more than two test patterns are used.

[0040] When the read pointer again indicates that all values have been read and compared (e.g., Rptr=511), the state machine moves from State 4 to the Done State (506). At this point in the BIST process all appropriate error flags should be set in the volatile memory cells (e.g., see Figs. 3 and 4). Bypassing is then enabled, e.g., by applying a high value to signal BIST_done in Fig. 4. In some embodiments, the BIST control circuit supplies a status signal (e.g., BIST_done) to the configuration logic indicating that the BIST procedure is complete. In other embodiments, the configuration control circuit includes a counter timer that defers the loading of the configuration data until sufficient time has passed to complete the BIST procedure. (In the present example, with $(16 + 512)$ clock cycles for each of two test patterns, a counter timer that counts to 1100 clock cycles can be used, for example.) In either case, after completion of the BIST procedure the configuration process continues, and configuration data is loaded into the RAM circuit, bypassing any defective column in the array.

[0041] Note that some arrays might include more than one defective column. This drawback can be alleviated by providing additional redundant columns, e.g., as in the embodiment illustrated in Fig. 6.

[0042] Fig. 6 is a block diagram of a second PLD including two RAM circuits that are automatically tested and repaired during the PLD configuration process according to an embodiment of the present invention. In the pictured embodiment, each RAM circuit 601a, 601b includes four redundant columns 609a, 609b, respectively.

[0043] Note that in Fig. 6, the columns of RAM cells lie horizontally. The words "columns" and "rows" as applied to physical orientation can easily be interchanged by simply (for example) rotating a figure, PLD, or RAM circuit by 90

degrees. Therefore, the words "column" and "row" as used herein are not intended to imply any particular physical orientation. Further, the word "column" as used herein refers to a linear grouping of RAM cells oriented along either a bit line or a word line of the RAM array, while the word "row" refers to a linear grouping of RAM cells oriented orthogonally to a related column. In the exemplary circuits described herein, columns of RAM cells lie along bit lines of the RAM array. However, the invention can also be applied to circuits wherein the columns of RAM cells are oriented along word lines of the RAM array.

[0044] The PLD of Fig. 6 includes first and second RAM circuits 601a and 601b, a BIST control circuit 602, first and second routing circuits 603a and 603b coupled to first and second read/write data ports 607a and 607b, a configuration control circuit 604, a configuration data port 605, and, optionally, additional PLD logic 606. RAM circuits 601a, 601b each include an array 613a, 613b comprising n columns of RAM cells, where n is a positive integer, and four redundant columns 609a, 609b of RAM cells. The PLD of Fig. 6 is similar to that of Fig. 1, but differs from the PLD of Fig. 1 in certain ways that are now described.

[0045] RAM circuits 601a and 601b can be considered either as a single RAM circuit, or as two separate but similar RAM circuits. If considered as a single RAM circuit, the circuit includes two groups 609a and 609b of four redundant columns each. The groups of redundant columns are equally spaced within the RAM circuit. If considered as two separate RAM circuits, each RAM circuit 601a, 601b includes four redundant columns. In the pictured embodiment, the four redundant columns are grouped at the bottom edge of each RAM circuit (i.e., the edge towards the bottom of the figure). Each RAM circuit 601a, 601b is controlled in parallel by BIST control circuit 602. Thus, BIST control circuit 602 can simultaneously test both arrays, setting error flags within

each RAM circuit as appropriate for any defective columns located.

[0046] In one embodiment, a PLD includes many RAM logic blocks. Each block RAM (BRAM) includes two similar RAM circuits, and each RAM circuit includes a first array of 128 rows and 72 columns, plus four redundant columns. In addition, a ring of dummy RAM cells is included around the periphery of each RAM circuit, to duplicate for the peripheral RAM cells the proximity effects experienced by the RAM cells located away from the periphery of the RAM circuit. In this embodiment, two sets of row decoders and drivers are located between the two similar RAM circuits, to balance the word line loading. The configuration control circuit for the BRAM and the BIST control circuit are also located between the two similar RAM circuits.

[0047] Fig. 6 also illustrates how a defective column group in each of the two arrays is bypassed. A column group can include, for example, a group of four adjacent columns that use shared sense amplifier circuitry. For example, in Fig. 6, blocks 612a and 612b represent rows of sense amplifiers, each sense amplifier being associated with four columns of RAM cells. Thus, a defect might be detected and isolated only to the extent that it is known to be somewhere in the circuitry associated with a particular sense amplifier.

[0048] For example, in Fig. 6, the circuitry in RAM circuit 601a that is identified as being defective includes four columns of RAM cells 614a and one associated sense amplifier circuit. As in Fig. 2, the defective circuitry is denoted by a cross-hatched pattern. The dotted lines through routing circuit 603a illustrate how the defective circuitry 614a is bypassed, e.g., using a circuit similar to that shown in Fig. 3.

[0049] Similarly, the circuitry in RAM circuit 601b that is identified as being defective includes four columns of RAM cells 614b and one associated sense amplifier circuit.

Again, the defective circuitry is denoted by a cross-hatched pattern. The dotted lines through routing circuit 603b illustrate how the defective circuitry 614b is bypassed, e.g., using a circuit similar to that shown in Fig. 3.

[0050] Note that a defect in either the RAM cell column or the sense amplifier circuits can be localized and repaired, as illustrated in Fig. 6.

[0051] Fig. 7 illustrates the steps of an exemplary method of configuring a PLD, according to another aspect of the present invention. The PLD includes a RAM circuit that includes both a first array of rows and columns of RAM cells and a redundant column of the RAM cells.

[0052] In step 701, configuration of the PLD is initiated, e.g., on power up of the device. In step 702, a built in self test (BIST) procedure is initiated on a RAM array included in the PLD. In some embodiments, this step is performed by a configuration control circuit, which provides an enable value on a BIST enable signal directed to a BIST control circuit.

[0053] During the BIST procedure, if errors are found (decision 703) one or more error flags are set (step 706) to indicate the defective column or columns. These error flags can be set, for example, by storing an error value in a volatile memory circuit associated with the defective column. In step 707, the configuration process for the PLD continues, with configuration data written to the RAM circuit being automatically shunted to bypass the defective columns and use the redundant columns.

[0054] If no defective columns are found during the BIST procedure, the configuration process for the PLD continues, with configuration data written to the RAM circuit being written to the standard columns (step 704). The redundant columns are not used.

[0055] After configuration of the PLD is complete (step 705), the PLD typically signals completion of the

configuration process, e.g., by providing a high value on a DONE signal.

[0056] Note that the BIST procedure described above tests the RAM circuit and repairs defective columns in the RAM circuit without any intervention on the part of either the user or the PLD manufacturer.

[0057] Fig. 8 illustrates the steps of a method of detecting and repairing defective columns in a RAM array, according to another embodiment of the invention. In the embodiment of Fig. 8, the BIST procedure is initiated independently of the configuration sequence for the PLD, e.g., on a previously-configured PLD.

[0058] In step 801, a built in self test (BIST) procedure is initiated on a RAM array included in the PLD. In some embodiments, this step is performed by a configuration control circuit, which provides an enable value on a BIST enable signal directed to a BIST control circuit. In other embodiments, the procedure is initiated by a control signal provided by a user, and not as part of a configuration process for the PLD. Because the BIST procedure is initiated by a user control signal, the PLD need not be powered down or reconfigured to perform the self-test and repair.

[0059] During the BIST procedure, if errors are found (decision step 802) one or more error flags are set (step 804) to indicate the defective column or columns. These error flags can be set, for example, by storing an error value in a volatile memory circuit associated with the defective column. User operation is then resumed (step 805). The PLD logical circuits include logic that automatically shunts read data from, and write data to, redundant columns when an associated error flag is set.

[0060] If no defective columns are detected in the RAM array during the BIST procedure (decision step 802), normal user operation is resumed (step 803). The redundant columns are not used.

[0061] Those having skill in the relevant arts of the invention will now perceive various modifications and additions that can be made as a result of the disclosure herein. For example, the above text describes the circuits of the invention in the context of programmable logic devices (PLDs) such as field programmable gate arrays (FPGAs). However, the circuits of the invention can be implemented in any programmable logic device that includes volatile memory cells. As another example, memory array sizes used herein are purely exemplary, and the concepts of the invention can be applied to memory arrays of any size.

[0062] Further, transistors, configuration control circuits, routing circuits, BIST control circuits, BIST test patterns, AND gates, NAND gates, XNOR gates, state machines, sense amplifiers, inverters, memory cells, volatile memory cells, volatile memory circuits, RAM circuits, RAM cells, multiplexers, and other components other than those described herein can be used in implementing the invention. Active-high signals can be replaced with active-low signals by making straightforward alterations to the circuitry, such as are well known in the art of circuit design. Logical circuits can be replaced by their logical equivalents by appropriately inverting input and output signals, as is also well known.

[0063] Moreover, some components are shown directly connected to one another while others are shown connected via intermediate components. In each instance the method of interconnection establishes some desired electrical communication between two or more circuit nodes. Such communication can often be accomplished using a number of circuit configurations, as will be understood by those of skill in the art.

[0064] Accordingly, all such modifications and additions are deemed to be within the scope of the invention, which is to be limited only by the appended claims and their equivalents.